IN THE CLAIMS

1. (Currently Amended) A computer system comprising:

an application memory organized as a plurality of cache lines, each cache line being identified by an address;

a FIFO buffer for storing a plurality of cache lines;

a central processing unit (CPU) for executing instructions stored in said application memory;

a state memory for storing the contents of the internal registers of said CPU;

a checkpoint controller for defining a series of repeating checkpoint cycles, said checkpoint controller having access to a plurality of registers in said CPU that define the state of that CPU at a point in each of said checkpoint cycles that is controllable by said checkpoint controller; ~~and~~

a memory controller for operating said application memory and said FIFO buffer, said memory controller receiving a cache line from said CPU in response to a write command specifying an address A in said application memory at which said cache line is to be stored, wherein a copy of said cache line as stored in said application memory at address A, is copied into said FIFO buffer upon receiving the first write command specifying A after the start of the current checkpoint cycle, said cache line received in said write command replacing the contents of A in said application memory; and

a plurality of reset flags, each reset flag corresponding to one of said cache lines in said application memory and having a first state indicating that the corresponding cache line in said application memory has been re-written since the start of the current checkpoint cycle and a second state indicating that the corresponding cache line has not been re-written since the start of said current checkpoint cycle, the state of said reset flags determining if said memory controller copies the contents of one of said cache lines into said FIFO buffer.

2. (Original) The computer system of Claim 1 wherein each checkpoint cycle comprises a computational phase and a checkpoint phase and wherein said checkpoint

controller during said checkpoint phase causes said CPU to write back to said memory all dirty cache lines; and

to store internal registers defining the state of said CPU in said state memory.

3. (Original) The computer system of Claim 2 wherein said checkpoint controller empties the contents of said FIFO buffer at the end of each checkpoint phase if no error has been detected by the end of said checkpoint phase.

4. (Original) The computer system of Claim 1 wherein said checkpoint controller, in response to a determination that a processing error has occurred, copies the contents of said FIFO buffer into said application memory, causes said CPU to copy the contents of said state memory into said CPU's internal registers, and restarts said computer system.

5. (Original) The computer system of Claim 4 wherein said checkpoint controller causes said computer system to be reconfigured before restarting said computer system.

6. (Original) The computer system of Claim 1 wherein said application memory comprises a fault tolerant memory.

7. Cancel claim 1.

8. (Original) The computer system of Claim 1 further comprises first and second vectors of flags, each vector being sufficient to provide said plurality of reset flags, said first vector supplying said reset flags in every other checkpoint cycle and said second vector supplying said reset flags in said remaining checkpoint cycles, said vector that is not supplying said reset flags in any give cycle being set to said second state during that cycle.

9. (Original) The computer system of Claim 1 wherein said memory controller causes a copy of said cache line as stored in said application memory at an address A to be copied into said FIFO buffer upon receiving the first read command specifying A after the start of the current checkpoint cycle if said read command assigns exclusive ownership of said cache line to said CPU.